

第一章 操作系统引论

一、名词解释

1. 操作系统 操作系统是管理计算机硬件与软件资源的系统软件
2. 预输入 通过提前将输入数据加载到高速缓存,减少CPU等待时间,优化I/O性能
3. 分布式操作系统 利用软件系统方式构建在计算机网络上的一种处理机系统
4. 实时操作系统 专为时间关系型任务设计的操作系统,采用抢占式任务优先级调度机制
5. 互斥共享 多个进程或线程在访问一临界资源时,通过互斥机制,确保同一时间只有一个执行单元可以访问该资源。

二、选择题

1. 在计算机系统中配置操作系统的主要目的是 (A)。操作系统的主要功能是管理计算机系统 (B)。

- A: (1) 增强计算机系统的功能; (2) 提高系统资源的利用率;
(3) 提高系统的运行速度; (4) 合理组织系统的工作流程,以提高系统吞吐量。?
- B: (1) 程序和数据; (2) 进程; (3) 资源; (4) 作业; (5) 软件; (6) 硬件。

2. 操作系统是一种 (A), 它负责为用户和用户程序完成所有 (B) 的工作, (C) 不是操作系统关心的主要问题。

- A: (1) 应用软件; (2) 系统软件; (3) 通用软件; (4) 软件包。
- B: (1) 与硬件无关并与应用无关; (2) 与硬件相关而与应用无关;
(3) 与硬件无关而与应用相关; (4) 与硬件相关并与应用相关。
- C: (1) 管理计算机裸机; (2) 设计、提供用户程序与计算机硬件系统的接口;
(3) 管理计算机中的信息资源; (4) 高级程序设计语言的编译。

3. 在设计分时操作系统时, 首先要考虑的是 (A); 在设计批处理操作系统时, 首先要考虑的是 (B); 在设计实时操作系统时, 首先要考虑的是 (C)。

- A, B, C: (1) 灵活性和可适应性; (2) 交互性和响应时间;
(3) 周转时间和系统吞吐量; (4) 实时性和可靠性。

4. 从下面关于操作系统的论述中, 论述正确的是 (A)

- 1) 对批处理作业, 必须提供相应的作业控制信息。
- 2) 对于分时系统, 不一定全部提供人机交互功能。
- 3) 从响应角度看, 分时系统与实时系统的要求相似。
- 4) 采用分时操作系统的计算机系统中, 用户可以独占计算机操作系统中的文件系统。
- 5) 从交互角度看, 分时系统与实时系统相似。

5. 在下列系统中, (A) 是实时信息系统, (B) 是实时控制系统。

- A, B: (1) 计算机激光照排系统; (2) 民航售票系统; (3) 办公自动化系统;
(4) 计算机辅助设计系统; (5) 火箭飞行控制系统。

三、填空题

1. 一个完整的计算机系统包括 **硬件** 和 **软件** 两部分。
2. 只有计算机硬件而没有任何软件支持的计算机称为 **裸机**。
3. 计算机系统资源包括 **硬件资源** 和 **软件资源** 两部分。
4. 单道批处理系统的特点包括单道性、**顺序性** 和自动性。
5. 操作系统的特征包括并发性、**共享**、**异步** 和虚拟性。
6. 处理机管理主要包括 **进程控制**、进程同步、进程通信和 **调度**。

7. 操作系统为用户提供的接口主要有命令接口、~~命令接口~~和图形用户接口。
8. MS-DOS 是 () 操作系统, Windows 95 是 () 操作系统, Windows XP 是 () 操作系统。
~~单用户多任务~~ ~~单用户多任务~~ ~~多用户多任务~~

四、判断题

1. (X) 用户用 C 语言编写的工资管理程序属于系统软件。
2. (✓) 在多道批处理系统中, 内存中可同时驻留多道程序, 这些程序可以并发执行。
3. (X) 分时系统中, 时间片设置得越小, 则平均响应时间越短。
4. (✓) 实时系统在响应时间、可靠性及交互作用能力等方面一般都比分时系统要求高。
5. (X) 在分时系统中, 为使多个用户能够同时与系统交互, 最关键的问题是系统能及时接收多个用户的输入。
6. (X) 在软实时系统中, 系统必须满足任务对截止时间的要求, 否则可能出现不可预期的结果。
7. (✓) 可靠性是实时系统的一大特点。
8. (✓) 并行性是指两个或多个事件在同一时刻发生, 强调时间点。
9. (X) 虚拟存储器是指从物理上扩充内存的容量。
10. (X) 微内核操作系统结构是现代操作系统普遍采用的结构。

第二章 进程的描述与控制

一、名词解释

1. 进程上下文 ~~进程上下文是指进程执行时所需的全部状态信息~~
2. 直接制约 ~~指进程或线程之间因共享资源或协作关系而产生的相互限制~~
3. 临界区 ~~进程中访问共享资源的代码段, 必须互斥执行。~~
4. 进程同步 ~~协调新进程的执行顺序, 以确保它们正确地访问共享资源或完成协作任务~~
5. 内核线程 ~~由操作系统内核直接管理的线程, 其创建、调度、销毁均由内核完成~~

二、选择题

1. 进程的三个基本状态是 (A)、(B)、(C)。由 (A) 到 (B) 是由进程调度所引起的; 由 (B) 到 (C) 是正在执行的进程发生了某事件, 使之无法继续执行而引起的。
A, B, C: (1) 挂起; (2) ~~C~~ 阻塞; (3) ~~A~~ 就绪; (4) ~~B~~ 执行; (5) 睡眠; (6) 完成; (7) 僵尸。
2. 在一个单处理机系统中, 存在 7 个进程, 则最多有 (A) 个进程处于就绪队列, (B) 个进程处于阻塞状态。
A, B: (1) 7; (2) ~~AB~~ 6; (3) 5; (4) 4; (5) 3; (6) 2; (7) 1; (8) 0。
3. 在分时系统中, 导致进程创建的典型事件是 (A); 在批处理系统中, 导致进程创建的典型事件是 (B); 由系统专门为运行中的应用进程创建新进程的事件是 (C)。在创建进程时, (D) 不是创建所必需的步骤。
A: (1) 用户注册; (2) ~~✓~~ 用户登录; (3) 用户记账; (4) 用户通信。
B: (1) 作业录入; (2) ~~✓~~ 作业调度; (3) 进程调度; (4) 中级调度。
C: (1) 分配资源; (2) 进程通信; (3) 共享资源; (4) ~~✓~~ 提供服务。
D: (1) 为进程建立 PCB; (2) 为进程分配内存等资源; (3) ~~✓~~ 为进程分配 CPU; (4) 将进程插入就绪队列。

4. 在生产者—消费者问题中，应设置互斥信号量 mutex、资源信号量 full 和 empty。它们的初值应分别是 (A)、(B) 和 (C)。

A, B, C: (B) 0; (A) 1; (C) -1; (4) -n; (5) +n; (6) n+1.

5. 下面的叙述中，正确的一条是 (B)。

- (1) 一个进程的状态发生变化总会引起其他一些进程的状态发生变化。
- (2) 进程被挂起后，状态变为阻塞状态。
- (3) 信号量的初值不能为负数。
- (4) 在进程对应的代码中使用 wait、signal 操作后，可以防止系统死锁。

三、填空题

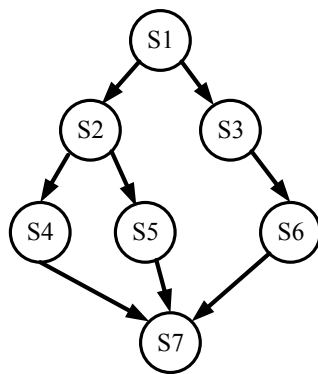
- 1. 进程特征包括其结构特征、~~动态性~~ 并发性、~~异步性~~ 独立性和 ~~结构性~~。
- 2. 若进程的用户程序段在执行时，该进程处于 ~~运行态~~，也称为 ~~用户态~~。
- 3. PCB 中包含有进程的描述信息、~~控制块~~ 以及 ~~运行信息~~，是进程动态特征的集中反映。
- 4. 进程空间分为用户空间和 ~~内核空间~~。
- 5. 在比较进程和程序时，进程是 ~~动态~~ 的概念，而程序是 ~~静态~~ 的概念。
- 6. 由共享公有资源而造成的对并发进程执行速度的制约称为 ~~互斥~~。
- 7. 进程同步机制应遵循的原则包括 ~~空闲让进~~ ~~让权等待~~ ~~有限等待~~ 和 ~~死锁避免~~。
- 8. 进程通信包括 ~~共享~~、~~消息传递~~ 和 ~~管道通信~~ 三种方式。
- 9. 线程的基本状态包括 ~~就绪~~ 状态、~~运行~~ 状态和 ~~阻塞~~ 状态。
- 10. 在间接通信方式中，进程之间的通信需要通过 ~~缓冲区~~ 来暂时存放消息。

四、判断题

- 1. (X) 操作系统的一个重要概念是进程，不同进程所执行的代码也不同。~~可以相同~~
- 2. (X) 操作系统通过 PCB 来控制和管理进程，用户进程可从 PCB 中读出与本身运行状态相关的信息。~~用户无权~~
- 3. (✓) 当进程由执行状态变为就绪状态时，CPU 现场信息必须被保存在 PCB 中。
- 4. (X) 当进程申请 CPU 得不到满足时，它将处于阻塞状态。
- 5. (✓) 管程每次只允许一个进程进入。
- 6. (X) wait、signal 操作可以解决一切互斥问题。
- 7. (X) 程序的顺序执行具有不可再现性。
- 8. (X) 临界区是指进程中用于实现进程同步的那段代码。
- 9. (✓) 当某进程处于静止就绪状态时，该进程在外存。
- 10. (X) 原语可以并发执行，并且其执行过程可以被中断。

五、问答题

- 1. 在操作系统中，何为进程“忙等”和“饥饿”？它们分别有什么缺点？
- 2. 试写出下图所示各个进程的前驱关系，并使用信号量机制设计程序执行所有进程。



3. 桌上有个能盛下 5 个水果的空盘子。父亲不停地向盘中放苹果或桔子，儿子不停地从盘中取出桔子，女儿不停地从盘中取出苹果。规定 3 人不能同时从盘中取放水果。试用信号量实现 3 人代表的进程之间的同步。

4. 有 3 个进程 PA, PB 和 PC 协作解决文件打印问题，PA 将文件记录从磁盘中读入内存的缓冲区 1，每执行一次都一个记录；PB 将缓冲区 1 的内容复制到缓冲区 2，每执行一次复制一个记录；PC 将缓冲区 2 的内容打印出来，每次执行打印一个记录，缓冲区的大小与记录大小一样。请使用 PV 操作实现进程的同步。

5. 嗜睡理发师问题：一个理发店有一个有 N 张沙发的等候室和一个放有 1 张理发椅的理发室组成，没有顾客理发师理发师便去睡觉，当一个顾客走进理发店时，如果所有的沙发都已被占用，他便离开理发店；否则，如果理发师正在为其他顾客理发，该顾客就找一张空沙发坐下等待，如果理发师因无顾客正在睡觉，由新到的顾客唤醒理发师理发，在理发完成后，顾客必须付费方式收费后才能离开理发店，试用 PV 操作实现此同步问题。

第三章 处理机调度与死锁

一、名词解释

1. 作业 作业是指用户提交给计算机系统的一个完整的计算任务，通常包括程序、数据及控制信息。
2. 处理机调度 操作系统决定哪道进程或线程获得 CPU 使用权的过程，目的是提高 CPU 利用率，保证公平性和响应速度。
3. 周转时间 作业从提交到完成所经历的时间，包括等待时间和执行时间，周转时间 = 完成时间 - 提交时间。
4. 死锁 多个进程因竞争资源而陷入相互等待的状态，导致所有进程无法继续执行。
5. 临时性资源 使用后就会消失的资源，无法重复利用，通常由进程动态创建和销毁。

二、选择题

1. 作业调度是从处于 (A) 状态的队列中选取作业投入运行，(B) 是指作业进入系统到作业完成所经过的时间间隔，(C) 算法不适合作业调度。

A: (1) 运行；(2) 提交；(3) 后备；(4) 完成；(5) 阻塞；(6) 就绪。

B: (1) 响应时间；(2) 周转时间；(3) 运行时间；(4) 等待时间；(5) 触发时间。

C: (1) 先来先服务；(2) 短作业优先；(3) 最高优先权优先；(4) 时间片轮转。

2. 如果为每一个作业只建立一个进程，则为了照顾短作业用户，应采用 (A)；为照顾紧急作业的用户，应采用 (B)；为能实现人机交互作用应采用 (C)；为了兼顾短作业和长时间等待的作业应采用 (D)；为了使短作业、长作业及交互型作业用户都比较满意应采用 (E)；为了使作业的平均周转时间最短应采用 (F) 算法。

A, B, C, D, E, F: (1) FCFS 调度算法；(2) 短作业优先；(3) 时间片轮转法；(4) 多

A
F

C

E

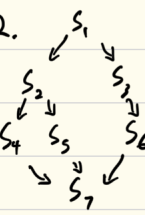
五. 简答题

2. 阻塞: 进程在循环中持续检查某条件, 不释放CPU让其他进程运行。

饥饿: 进程长期得不到所需资源, 导致其无法执行。

缺点: ①CPU资源浪费 ②优先级反转 (高优先级等待低优先级) ③单CPU可能死锁

①资源利用率下降 ②系统公平性差 ③响应时间不可预测

2.  前趋关系: $S_1 \xrightarrow{a} S_2$, $S_1 \xrightarrow{b} S_3$, $S_2 \xrightarrow{c} S_4$, $S_2 \xrightarrow{d} S_5$, $S_3 \xrightarrow{e} S_6$, $S_4 \xrightarrow{f} S_7$, $S_5 \xrightarrow{g} S_7$, $S_6 \xrightarrow{h} S_7$

```
main() {
    Semaphore a, b, c, d, e, f, g, h; a.Value = 0; b.Value = 0; c.Value = 0; d.Value = 0; e.Value = 0; f.Value = 0; g.Value = 0; h.Value = 0;
    Cobegin
        { S1; Signal(a); Signal(b); } { wait(a); S2; Signal(c); Signal(d); } { wait(b); S3; Signal(e); } { wait(e); S6; Signal(f); }
        { wait(d); S5; Signal(g); } { wait(f); S7; }
    Corend
}
```

3.

```
main() {
    Semaphore mutex = 1; Semaphore empty = 5; Semaphore apple = 0; Semaphore orange = 0;
    Void father() {
        While(true) {
            fruit = 苹果或橙子;
            wait(empty);
            wait(mutex);
            if (fruit == 苹果) {
                Signal(apple);
            } else {
                Signal(orange);
            }
            Signal(mutex);
        }
    }
    Void son() {
        While(true) {
            wait(orange);
            wait(mutex);
            取走橙子;
            Signal(empty);
            Signal(mutex);
        }
    }
    Void daughter() {
        While(true) {
            wait(apple);
            wait(mutex);
            取走苹果;
            Signal(empty);
            Signal(mutex);
        }
    }
    Cobegin { father(); son(); daughter(); }
    Corend
}
```

4.

```
main() {
    Semaphore empty1 = 1; empty2 = 1; full1 = 0; full2 = 0;
    PAC() {
        While(true) {
            从缓冲区1中取数据;
            P(empty1);
            取入数据;
            V(full1);
        }
    }
    PBC() {
        While(true) {
            P(full1);
            从缓冲区1中取数据;
            V(empty1);
            P(empty2);
            将数据存入缓冲区2;
            V(full2);
        }
    }
    PC() {
        While(true) {
            P(full2);
            将数据从缓冲区2取出;
            V(empty2);
        }
    }
    Cobegin { PAC(); PBC(); PC(); }
    Corend
}
```

```

5. main() {
    Semaphore sofa = 1; payment = 0; barber_ready = 1; customer_ready = 0; mutex = 1;

    barber() {
        P(customer_ready);
        P(mutex);
        叫号;
        V(barber_ready);
        V(sofa);
        V(mutex);
        剪头;
        P(payment);
    }

    Customer() {
        P(mutex);
        if (sofa.value == 0) {
            V(mutex);
            离开;
        } else {
            P(sofa);
            V(customer_ready);
            V(mutex);
            P(barber_ready);
            去剪头;
            V(payment);
        }
    }

    cobegin {
        Customer();
        barber();
    }
    coend
}

```


级反馈队列调度算法；(5)^B基于优先权的剥夺调度算法；(6)^D高响应比优先。

3. 系统产生死锁是指(A)。产生死锁的基本原因是(B)和(C)，产生死锁的四个必要条件是互斥条件、(D)、不剥夺条件和(E)。

A: (1) 系统发生重大故障；(2) 若干进程同时处于阻塞状态；(3) 若干进程正在等待永远不可能得到的资源；(4) 请求的资源数大于系统提供的资源数；(5) 若干进程等待被其他进程所占用而又不可能被释放的资源。

B: (1) 资源分配不当；(2) 系统资源不足；(3) 作业调度不当；(4) 资源的独占性。

C: (1) 进程推进顺序不当；(2) 进程调度不当；(3) 系统中进程太多；(4) CPU 运行太快。

D: (1) 请求和阻塞条件；(2) 请求和释放条件；(3) 请求和保持条件；(4) 释放和阻塞条件；(5) 释放和请求条件。

E: (1) 线性增长条件；(2) 环路条件；(3) 无序释放条件；(4) 有序释放条件；(5) 无序请求条件。

4. 下述解决死锁的方法中，属于死锁预防策略的是(A)，属于死锁避免策略的是(B)。

A, B: (1) 银行家算法；(2) 资源有序分配法；(3) 资源分配图化简法；(4) 撤消进程法。

5. 死锁的预防是通过破坏产生死锁的四个必要条件来实现的。下列方法中，(A)破坏了“请求与保持”条件，(B)破坏了“循环等待”条件。

A, B: (1) 银行家算法；(2) 一次性分配策略；(3) 资源有序分配策略；(4) SPOOLing 技术。

三、填空题

1. 作业的输入方式包括联机、脱机、直接和假脱机。
2. 作业在其生存期间会经历提交、后备、执行以及完成等状态。
3. 处理机调度的类型分为高级调度、中级调度和低级调度，其中，中级调度又称为预调度和内调调度。
4. 优先数的确定分为静态和动态两种。
5. 根据响应时间分类，可以将实时系统分为硬实时和软实时和一般实时系统。
6. 死锁的处理方法包括预防、避免、检测和解除。

四、判断题

1. (X) 系统处于不安全状态必然会导致死锁。
2. (✓) 竞争可同时共享的资源，不会导致系统进入死锁状态。
3. (X) 计算作业的优先权应高于 I/O 型作业的优先权。
4. (X) 资源要求多的作业，其优先权应高于资源要求少的作业。
5. (✓) 在动态优先权时，随着进程执行时间的增加，其优先权降低。
6. (✓) 预防死锁设置的限制条件比避免死锁严格，不利于进程的并发执行。
7. (X) 实时系统的输出结果的正确性仅仅依赖于结果的正确性。
8. (X) 在多级反馈队列调度算法中，优先权越高的队列，其执行的时间片越短。
9. (X) 响应比是等待时间与要求服务的时间之比。
10. (✓) 作业的概念一般用于早期批处理系统和现在的大型机、巨型机系统中，对于微机和工作站系统一般不使用作业的概念。

五、问答题

1. 假设一个系统中有 5 个进程，它们的到达时间和服务时间如下表所示，忽略 IO 及其他系统开销时间，若分别按照先来先服务 (FCFS)、非抢占和抢占的短作业优先 (SJF)、高响应

比优先（HRRN）和时间片轮转（RR，时间片为 1）算法进行 CPU 调度，请给出各个进程的完成时间、周转时间、带权周转时间、平均周转时间和平均带权周转时间。

进程	到达时间	服务时间
A	0	3
B	2	6
C	4	4
D	6	5
E	8	2

2. 若有 3 个周期性任务，任务 A 要求每 20ms 执行一次，执行时间为 10ms；任务 B 要求每 50ms 执行一次，执行时间为 10ms；任务 C 要求每 50ms 执行一次，执行时间为 15ms。应如何按最低松弛度优先算法对它们进行 CPU 调度？

3. 在银行家算法中，若出现下面的资源分配情况：

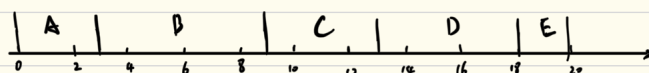
进程	Max	Need	Available
P0	0 0 4 4	0 0 1 2	1 6 2 2
P1	2 7 5 0	1 7 5 0	
P2	3 6 10 10	2 3 5 6	
P3	0 9 8 4	0 6 5 2	
P4	0 6 6 10	0 6 5 6	

- (1) 请计算分配矩阵的值，并判断该状态是否安全？
- (2) 若进程 P2 提出请求 Request(1,2,2,2)，系统能否将资源分配给它？
- (3) 如果系统立即满足 P2 的上述请求，请问系统是否立即进入死锁状态？

1.

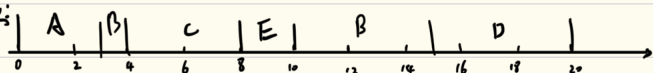
进程	到达时间	服务时间
A	0	3
B	2	6
C	4	4
D	6	5
E	8	2

先来先服务:

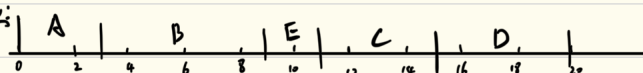


	A	B	C	D	E	平均
完成时间	3	9	13	18	20	
周转时间	3	7	9	12	12	8.6
带权周转时间	1	$\frac{7}{6}$	$\frac{9}{4}$	$\frac{12}{5}$	$\frac{12}{2}$	2.56

抢占式作业:



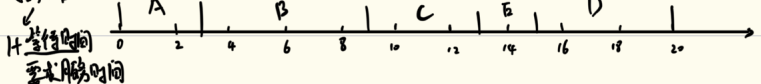
非抢占式作业:



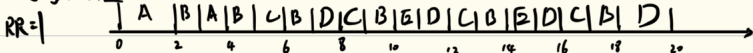
	A	B	C	D	E	平均
完成时间	3	15	8	20	10	
周转时间	3	13	4	14	2	7.2
带权周转时间	1	$\frac{13}{6}$	1	$\frac{14}{5}$	1	1.59

	A	B	C	D	E	平均
完成时间	3	9	15	20	11	
周转时间	3	7	11	14	3	7.6
带权周转时间	1	$\frac{7}{6}$	$\frac{11}{4}$	$\frac{14}{5}$	$\frac{3}{2}$	1.84

高响应比:



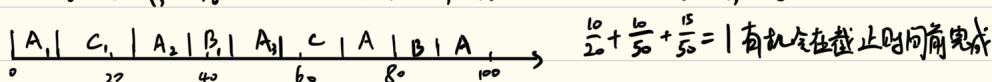
时间片轮转:



	A	B	C	D	E	平均
完成时间	3	9	13	20	15	
周转时间	3	7	9	14	7	8
带权周转时间	1	$\frac{7}{6}$	$\frac{9}{4}$	$\frac{14}{5}$	$\frac{7}{2}$	2.14

	A	B	C	D	E	平均
完成时间	4	18	17	20	15	
周转时间	4	16	13	14	7	10.8
带权周转时间	$\frac{4}{3}$	$\frac{8}{3}$	$\frac{13}{4}$	$\frac{14}{5}$	$\frac{7}{2}$	2.71

2. A: 20ms 执行一次, 执行 10ms B: 50ms 执行一次, 执行 10ms C: 50ms 执行一次, 执行 15ms



0:

A: 20-10-0=10 B: 50-10-0=40 C: 50-15-0=35 70: A: 80-10-70=0 B: 60-10-70=20

25:

A: 40-10-25=5 B: 50-10-25=15 80: A: 100-80-10=10 B: 100-10-80=10

55:

B: 100-55-10=35 C: 100-55-15=30

3.

进程	Allocation	Need	available
P ₀	0 0 3 2	0 0 1 2	1 6 2 2
P ₁	1 0 0 0	1 7 5 0	
P ₂	1 3 5 4	2 3 5 6	
P ₃	0 3 3 2	0 6 5 2	
P ₄	0 0 1 4	0 6 5 6	

(1)

进程	work	Allocation	Need	Work Allocation	Finish
P ₀	1 6 2 2	0 0 3 2	0 0 1 2	1 6 5 4	true
P ₃	1 6 5 4	0 3 3 2	1 6 5 2	1 9 8 6	true
P ₄	1 9 8 6	0 0 1 4	0 6 5 6	1 15 13 10	true
P ₁	1 15 13 10	1 0 0 0	1 7 5 0	2 15 13 10	true
P ₂	2 15 13 10	1 3 5 4	2 3 5 6	3 18 18 14	true

有一个安全序列 {P₀, P₃, P₄, P₁, P₂}

(2) $\text{request}_2(1, 2, 2, 2) \leq \text{Need}(2, 3, 5, 6)$

② $\text{request}_2(1, 2, 2, 2) \leq \text{available}(1, 6, 2, 2)$

③ 假设可以为 P₂ 分配资源

Available = (0, 4, 0, 0) Allocation = (2, 5, 7, 6)

Need₂ = (1, 1, 3, 4)

④ 进行安全性检测, 所有进程无法满足 $\text{Need}_i \leq \text{Available}$, 改为前为不安全状态, 系统不分配资源。

(3) 并没有马上进入死锁状态, 因为此时上述进程并没有申请新的资源, 并且得到资源而进入阻塞状态。只有当上述进程提出新的请求, 导致所有没执行完的多个进程因得不到资源而阻塞并形成循环等待时, 系统才进入死锁状态。